REMARKS

[0002] Applicant respectfully requests reconsideration and allowance of all of the claims of the application. The status of the claims is as follows:

- Claims 1, 3-6, 8-18, 20-23 and 26 are currently pending
- Claims 2, 7, 19, 24 and 25 are canceled herein
- No claims are withdrawn herein
- Claims 1, 3, 6, 21 and 26 are amended herein
- No new claims are added herein

[0003] Support for the amendments to claim 1 is found in the specification at least at paragraphs [0022], [0023], and [0027] and dependent claim 3. Claim 3 is amended herein to remove the portion now incorporated by independent claim 1. Support for the amendments to claim 6 is found in the specification at least at paragraph [0042]. Support for the amendments to claim 21 is found in the specification at least at paragraphs [0030]-[0031] and [0042]. Claim 26 is amended herein just to address formal matters, for example grammar.

Cited Documents

[0004] The following documents have been applied to reject one or more claims of the Application:

- **Zintel**: Zintel et al., U.S. Patent Application Publication No. 2002/0029256;
- **Saint**: Saint-Hilaire et al., U.S. Patent Application Publication No. 2003/0101294;
- **Slaughter**: Slaughter et al., U.S. Patent No. 6,643,650.

Zintel Fails to Anticipate Claims 1, 3-5 and 26

[0005] Claims 1, 3-5 and 26 stand rejected under 35 U.S.C. § 102(b) as allegedly being anticipated by Zintel. Applicant respectfully traverses the rejection.

Independent Claim 1

[0006] Applicant submits that the Office has not shown that Zintel anticipates this claim. Zintel does not disclose the following features of this claim, as amended (with emphasis added):

> one or more services executing in the device, the one or more services comprising:
>
> a data service, the data service being configured to produce **a customizable tag-based document that holds events which have been generated by the one or more services but not yet consumed by an application service, the customizable tag-based document holding a state of the one or more services, the data service being further configured to store the customizable tag-based document until requested by the application service;**

[0007] The primary cited reference, Zintel, does not disclose the claim language quoted above. Zintel does not disclose, "a customizable tag-based document that holds events which have been generated by the one or more services but not yet consumed by an application service, the customizable tag-based document holding a state of the one or more services, the data service being further configured to store the customizable tag-based document until requested by the application service." The Office addressed this claim language directly in its Response to Arguments portion of the Office Action dated 11/10/2009.

[0008] In that Response to Arguments, the Office states, "Zintel teaches one or more services comprising an information service…the information service being configured to produce a customizable tag-based document [0436-0452]." (Office Action, page 2). That section of Zintel describes:

[0442] f. The body of the message is composed from the list of properties stored within the event source structure:

[0443] i. Write the <propertyset>opening tag.

[0444] ii. Write the <propcount>n</propcount>tag. Where n is the number of total properties.

[0445] iii. For each property:

[0446] 1. Write the <property> opening tag.

[0447] 2. Write the <prop>opening tag, where prop is the name of the property.

[0448] 3. Write the < type>type< /type> tag, where type is the stringized type name of the property type.

[0449] 4. Write the property value.

[0450] 5. Write the </prop> closing tag.

[0451] 6. Write the </property> closing tag

[0452] iv. Write the </propertyset> closing tag

[0009] The Office appears to be equating the claimed "customizable tag-based document" with the message composed from a list of properties as described above in Zintel's paragraphs [00443]-[0452] where the properties are described in terms of tags (e.g., "opening tag").

[0010] Furthermore, the Office indicates that Zintel's message with property tags (which it equated with the claimed "customizable tag-based documents") "holds events

which have been generated by the one or more services but not yet consumed by an application service." (Office Action, page 2-3). The Office supports that assertion by saying that Zintel's "UPnP event notification is an XML message sent over HTTP/TCP to each and every subscriber to a particular UPnP service" and refers to paragraphs [0273-0277] of Zintel. (Office Action, page 2-3). The portion of Zintel that the Office relies upon for teaching that its message with property tags "holds events which have been generated by the one or more services but not yet consumed by an application service" is provided below for the convenience of the reader:

> [0273] What is an Event?
>
> [0274] Property change events are defined as any change in the value of a row of the Device State Table (DST) **230** (**FIG. 3**) for a service **210-217**. This change will be reflected as a property change notification. For example, if a "VCR" device has a "VCR Transport" service, one row in that service's DST may be TapeState and the value could be TapePresent. If the tape is ejected, the new value would be TapeAbsent. This state change would be reflected as a notification sent to all subscribers.
>
> [0275] What is a Notification?
>
> [0276] A UPnP event notification is an XML message sent over HTTP/TCP to each and every subscriber to a particular UPnP service. The content of the XML is defined below. The important contents of this message are the unique identifier for the subscription, the property name, new value, and property type.
>
> [0277] Notification Processing

[0011] It seems the Office is equating the claimed "events which have been generated by the one or more services but not yet consumed by an application service" with the "UPnP event notification...sent...to each and every subscriber to a particular

UPnP service," as described above by Zintel's paragraph [0276]. Unfortunately, there are at least two problems with the Office's interpretation here.

[0012] First, there is nothing indicating that the event-related portions of Zintel cited by the Office (paragraphs [0273-0277]) are related to Zintel's message with property tags (from paragraphs [0436-0452]). The claim language states "customizable tag-based documents [that] holds events...." Since the Office equates the claimed "documents" to Zintel's messages (from paragraphs [0436-0452]), there must be and the Office must show, at the very least, how Zintel's messages are associated with the event-related portions of Zintel cited by the Office (paragraphs [0273-0277]). The Office has not done that. Applicant submits that Zintel's messages have nothing to do with the event-related portions of Zintel cited by the Office (paragraphs [0273-0277]).

[0013] Second, Zintel is completely silent about whether its events ("UPnP event notification") are or are not yet "consumed by an application service," which is part of the recitation of this claim. Furthermore, the Office has not explained or shown how Zintel's events qualify as being "not yet consumed by an application service."

[0014] In addition, the Office indicates that Zintel's message with property tags (which it equated with the claimed "customizable tag-based documents") "holds the state [of] the one or more services." (Office Action, page 3). The Office supports that assertion by saying that Zintel's "UPnP events are mainly used for asynchronous notifications of state changes; paragraphs 0125 and 0294]." (Office Action, page 3). Those cited sections of Zintel say the following:

> **[0125]** Eventing, in the context of UPnP, is the ability for a device to initiate a connection at any time to one or more devices that have expressed a desire to receive events from the source device. Events are used to enable synchronization among multiple devices organized into a many to one relationship. UPnP events are mainly used for asynchronous notifications of state changes.
>
> **[0294]** Event—message generated when a change in a resource's state occurs.

[0015] It appears the Office is equating the "customizable tag-based document holding a state of the one or more services," as recited in part by claim 1, with the, "UPnP events are mainly used for asynchronous notifications of state changes," and "Event-message generated when a change in a resource's state occurs." (Zintel, paragraphs [0125] and [0294]).

[0016] However, the Office has failed to show how Zintel's message with property tags "hold[s] a state of the one or more services." Applicant submits that there is nothing indicating that these state-related portions of Zintel cited by the Office (paragraphs [0125] and [0294]) are related to Zintel's message with property tags (Zintel, paragraphs [0436-0452]). Since the Office equates the claimed "documents" to Zintel's messages (Zintel, paragraphs [0436-0452]), there must be and the Office must show, at the very least, how Zintel's messages are associated with the state-related portions of Zintel cited by the Office (paragraphs [0125] and [0294]). The Office has not done that. Applicant submits that Zintel's messages have nothing to do with the state-related portions of Zintel cited by the Office.

[0017] Consequently, Zintel does not disclose all of the elements and features of this claim. Accordingly, Applicant submits that Zintel does not anticipate this claim, and respectfully requests that the rejection of this claim be withdrawn.

Dependent Claims 3-5

[0018] Claims 3-5 ultimately depend from independent claim 1. As discussed above, claim 1 is not anticipated by the cited documents, and is therefore allowable over the cited documents. Therefore, claims 3-5 are also allowable over the cited documents of record for at least their dependency from an allowable base claim. These claims may also be allowable for the additional features that each recites.

Independent Claim 26

[0019] Applicant submits that the Office has not shown that Zintel anticipates this claim. Zintel does not disclose the following features of this claim, (with emphasis added):

> customizable tag-based messages **that hold events which have been generated by the one or more services but not yet consumed by an application service,** the data service being further configured to store the customizable tag-based message until requested by the application service, and further wherein **the customizable tag-based message holds the state of the one or more services.**

[0020] Claim 26 recites in part, "customizable tag-based messages that hold events which have been generated by the one ore or more services but not yet consumed by

an application service." The Office cites Zintel, paragraphs [0273]-[0277] as disclosing this feature. (Office Action, page 6-7.) Zintel describes:

> [0273] What is an Event?
>
> [0274] Property change events are defined as any change in the value of a row of the Device State Table (DST) **230** (**FIG. 3**) for a service **210-217**. This change will be reflected as a property change notification. For example, if a "VCR" device has a "VCR Transport" service, one row in that service's DST may be TapeState and the value could be TapePresent. If the tape is ejected, the new value would be TapeAbsent. This state change would be reflected as a notification sent to all subscribers.
>
> [0275] What is a Notification?
>
> [0276] A UPnP event notification is an XML message sent over HTTP/TCP to each and every subscriber to a particular UPnP service. The content of the XML is defined below. The important contents of this message are the unique identifier for the subscription, the property name, new value, and property type.
>
> [0277] Notification Processing

[0021] It appears the Office is equating the "events which have been generated," as recited in part by claim 26, with the "Events" as described by Zintel above in paragraph [0274]. However, nowhere does Zintel disclose, "customizable tag-based messages...hold events ...generated by one or more services but not yet consumed by an application service," as claim 26 recites in part. Instead, Zintel describes "property change events" which are "any change in the value of a row of the Device State Table...for a service," which then results in a "change notification." (Zintel, paragraph [0274]).

[0022] Claim 26 recites in part, "the customizable tag-based messages hold the state of the one or more services." The Office cites Zintel, paragraphs [0125] and [0294] as

disclosing this feature. (Office Action, page 7.) Paragraphs [0125] and [0294] are reproduced at paragraph [0014] of this response.

[0023] Zintel also describes, "Event-message generated when a change in a resource's state occurs," (Zintel, paragraph [0294]). However, nowhere does Zintel disclose, teach or suggest, "customizable tag-based messages hold the state of the one or more services," as claim 26 recites in part.

[0024] It seems the Office is equating the "UPnP events...mainly used for asynchronous notifications of state changes," of Zintel's paragraph [0125] with the claim 26 partial recitation of, "the customizable tag-based messages hold the state of one or more services."

[0025] However, the Office has failed to show how Zintel's messages "holds the state of the one or more services." Applicant further submits that there is nothing indicating that these state-related portions of Zintel cited by the Office (Zintel, paragraphs [0125] and [0294]) are related to Zintel's messages with property tags (Zintel, paragraphs [0436-0452]). Since the Office equates the claimed "customizable tag-based messages" to Zintel's messages (from paragraphs [0436-0452]), there must be and the Office must show, at the very least, how Zintel's messages are associated with the state-related portions of Zintel cited by the Office (Zintel, paragraphs [0125] and [0294]). The Office has not done that. Applicant submits that Zintel's messages have nothing to do with the state-related portions of Zintel cited by the Office.

[0026] Consequently, Zintel does not disclose all of the elements and features of this claim. Accordingly, Applicant submits that Zintel does not anticipate this claim, and respectfully requests that the rejection of this claim be withdrawn.

Claims 6, 8-12 and 14 Are Non-Obvious Over Zintel in view of Saint

[0027] Claims 6, 8-12 and 14 stand rejected under 35 U.S.C. § 103(a) as allegedly being obvious over Zintel in view of Saint. Applicant respectfully traverses the rejection.

Independent Claim 6

[0028] Applicant submits that amended independent claim 6 is not obvious in view of the combination of Zintel and Saint. Applicant submits that the combination of Zintel and Saint does not teach or suggest the following features of this claim, as amended (with emphasis added):

- **a display service that includes a cursor shape service** with a port identifiable by an identifier;
- **a customizable tag-based document that describes the shape of an on-screen cursor,** the customizable tag-based document **including image data specifying pixels that comprise the on-screen cursor.**

[0029] In short, instead of teaching the claimed "curser shape service", the closest cited art, Saint, teaches cursor positioning and scaling services. Defining the position and scale of a cursor is not the same as defining its shape.

[0030] Claim 6 recites in part, "a display service...that includes a cursor shape service." The Office admits that Zintel does not teach this feature (See Office Action, page 8). So the Office relies upon Saint and cites paragraphs [0144]-[0146] and paragraph [0155] as teaching this feature. (Office Action, page 8). Saint describes, "If the device has a display, these pointer primitives may have a different resolution that may not match the x and y of the display. In such instances, software residing on the

extended PC may be used to scale the pointer position to the display." (Saint, paragraph [0155]).

[0031] It appears the Office is equating the "cursor shape service" as recited in part by claim 6, with the "pointer primitives" and "scale [of] the pointer position" of Saint's paragraph [0155]. However, the claimed "cursor shape" is not the same as "pointer primitives" or "scale [of] the pointer position." Saint describes, "three pointer primitives...(down, move, up)...used is [sic] almost all operating systems. If the device has a display, these pointer primitives may have a different resolution that may not match the x and y of the display...software residing on the extended PC may be used to scale the pointer position to the display." (Saint, paragraphs [0152]-[0155]). Pointer primitives describe actions or movements of a pointer such as "down," "move," or "up."

[0032] Nowhere does Saint disclose, teach or suggest that its pointer primitives include "cursor shape" along with these movement descriptors. Furthermore, "scaling the pointer position to the display," as described by Saint in paragraph [0155], is also not the same as "cursor shape" as claim 6 recites in part, because scaling the pointer occurs when the pointer primitives (i.e. "down", "move" and "up") have a different resolution that does not match the x and y of the device's display. In contrast, the "cursor shape" recited in part by amended claim 6, is directed towards, "the shape of an on-screen cursor, the customizable tag-based document including image data specifying pixels that comprise the on-screen cursor."

[0033] Furthermore, Saint fails to disclose, teach or suggest a "customizable tag-based document," as recited in part by amended claim 6. Instead, Saint describes, "software residing on the extended PC may be used to scale the pointer position to the

display." (Saint, paragraph [0155]). The PC software used to "scale the pointer" of Saint is not the same as the "customizable tag-based document" of amended claim 6, because the "customizable tag-based document, "describes the shape on an on-screen cursor" and "include[s] image data specifying pixels that comprise the on-screen cursor." In contrast, Saint describes, "pointer primitives may have a different resolution that may not match the x and y of the display. In such instances, software residing on the extended PC may be used to scale the pointer position to the display." (Saint, paragraph [0155]). Scaling the pointer position to the display, as described by Saint, is not the same as "describ[ing] the shape" as claim 6 recites in part. Instead, scaling the pointer occurs when the pointer primitives (i.e. "down", "move" and "up") have a different resolution that does not match the x and y of the device's display. This is unlike amended claim 6 which recites in part, "customizable tag-based document including image data specifying pixels that comprise the on-screen cursor." Saint's "scaling the pointer...to match the x and y of the device's display" does not disclose, teach, or suggest the claimed "image data specifying pixels that comprise the on-screen cursor."

[0034] Consequently, the combination of Zintel and Saint does not teach or suggest all of the elements and features of this claim. Accordingly, Applicant respectfully requests that the rejection of this claim be withdrawn.

Dependent Claims 8-12 and 14

[0035] Claims 8-12 and 14 ultimately depend from independent claim 6. As discussed above, claim 6 is allowable over the cited documents. Therefore, claims 8-12 and 14 are also allowable over the cited documents of record for at least their

dependency from an allowable base claim. These claims may also be allowable for the additional features that each recites.

Claims 16 and 21 Are Non-Obvious Over Zintel in view of Slaughter

[0036] Claims 16 and 21 stands rejected under 35 U.S.C. § 103(a) as allegedly being obvious over Zintel in view of Slaughter. Applicant respectfully traverses the rejection.

Independent Claim 16

[0037] Applicant submits that the Office has not shown that independent claim 16 is obvious in view of the combination of Zintel and Slaughter. Applicant submits that the combination of Zintel and Slaughter does not teach or suggest the following features of this claim, (with emphasis added):

> requesting the service to **change a cursor shape**, the act of requesting **invoking a cursor shape service that changes the shape of the cursor**;

[0038] In short, instead of teaching the claimed "curser shape service" or "chang[ing] a cursor shape," the closest cited art, Saint, teaches cursor positioning and scaling services. Defining the position and scale of a cursor is not the same as defining its shape.

[0039] Claim 16 recites in part, "change a cursor shape," and "invoking a cursor shape service that changes the shape of the cursor." The Office states that, "Zintel and Slaughter does [sic] not specifically disclose requesting the service to change a cursor shape, the act of requesting invoking a cursor shape service that changes the shape of the cursor." (Office Action, page 12). Instead, the Office cites Saint, paragraphs [0144]-[0146], [0152] and [0155] as teaching this feature. (Office Action, page 12). Saint

describes, "various types of input, from no input at all to pen-input, keyboard, mouse, and button input. Each type of input will generally be handled by a respective input service, wherein the existence and capabilities of each input service will be described in that service's UPnP description information." (Saint, paragraph [0144]).

[0040] It appears the Office is equating the "cursor shape service" as recited in part by claim 16, with the "respective input service" of Saint's paragraph [0144]. This is not the same as the "respective input service" described by Saint in paragraph [0144], because Saint's "respective input service" does not disclose, teach or suggest the claimed "invoking a cursor shape service that changes the shape of the cursor."

[0041] Instead, Saint describes, "three pointer primitives...(down, move, up)...used is [sic] almost all operating systems. If the device has a display, these pointer primitives may have a different resolution that may not match the x and y of the display...software residing on the extended PC may be used to scale the pointer position to the display." (Saint, paragraphs [0152]-[0155]). Pointer primitives describe actions or movements of a pointer such as "down," "move," or "up." Saint's "pointer primitives" do not disclose, teach or suggest the claimed "change[s] the shape of the cursor.". Instead, pointer primitives only seem to address the "down", "move" and "up" aspect of the pointer.

[0042] Furthermore, Saint describes, "If the device has a display, these pointer primitives may have a different resolution that may not match the x and y of the display. In such instances, software residing on the extended PC may be used to scale the pointer position to the display." (Saint, paragraph [0155]). Scaling the pointer position to the display, as described by Saint, is not the same as "chang[ing] the shape of the cursor" as claim 16 recites in part. Instead, scaling the pointer occurs when the pointer

primitives (i.e. "down", "move" and "up") have a different resolution that does not match the x and y of the device's display.

[0043] Consequently, the combination of Zintel and Slaughter does not teach or suggest all of the elements and features of this claim. Accordingly, Applicant respectfully requests that the rejection of this claim be withdrawn.

Independent Claim 21

[0044] Applicant submits that the Office will be unable to show that independent claim 21 is obvious in view of the combination of Zintel and Slaughter. Applicant submits that the combination of Zintel and Slaughter does not teach or suggest the following features of this claim, as amended (with emphasis added):

- receiving a customizable tag-based document that **holds a state of the service representing the device** and the **input/output event that has been generated by the service** representing the device **but not yet consumed by an application;**
- **requesting the service to change a cursor shape**, the act of requesting invoking a cursor shape service that changes the shape of the cursor**, the cursor shape described by a customizable tag-based document**, the customizable tag-based document **including image data specifying pixels that comprise the cursor shape;**

[0045] Claim 21 is amended to recite in part, "receiving a customizable tag-based document that holds the state of the service representing the device," and "the input/output event that has been generated by the service…but not yet consumed by an application." The Office states, "As to claim 21, see the rejection to claim 16 above." (Office Action, page 13). In its rejection of claim 16, the Office cites Zintel, paragraph [0276] as teaching this feature. (Office Action, page 11). Zintel describes:

> [0276] A UPnP event notification is an XML message sent over HTTP/TCP to each and every subscriber to a particular UPnP service. The content of the XML is defined below. The important contents of this message are the unique identifier for the subscription, the property name, new value, and property type.

[0046] The primary cited reference, Zintel, does not disclose the claim language quoted above. Zintel does not disclose a "customizable tag-based document that holds a state of the service representing the device and the input/output event that has been generated by the service representing the device but not yet consumed by an application." The Office addressed this claim language directly in its Response to Arguments portion of the Office Action dated 11/10/2009.

[0047] In that Response to Arguments, the Office states, "Zintel teaches one or more services comprising an information service…the information service being configured to produce a customizable tag-based document [0436-0452]." (Office Action, page 2). That section of Zintel is reproduced in paragraph [0008] of this response.

[0048] The Office appears to be equating the claimed "customizable tag-based document" with the message composed from a list of properties as described above in Zintel's paragraphs [00443]-[0452] where the properties are described in terms of tags (e.g., "opening tag").

[0049] Furthermore, the Office indicates that Zintel's message with property tags (which it equated with the claimed "customizable tag-based documents"), "holds events which have been generated by the one or more services but not yet consumed by an application service." (Office Action, page 2-3). The Office supports that assertion by

saying that Zintel's "UPnP event notification is an XML message sent over HTTP/TCP to each and every subscriber to a particular UPnP service" and refers to paragraphs [0273-0277] of Zintel. (Office Action, page 2-3). The portion of Zintel that the Office relies upon for teaching that its message with property tags "holds events which have been generated by the one or more services but not yet consumed by an application service" is provided in paragraph [0020] of this response for the convenience of the reader:

[0050] It seems the Office is equating the claimed "events which have been generated by the one or more services but not yet consumed by an application service" with the "UPnP event notification...sent...to each and every subscriber to a particular UPnP service," as described above by Zintel's paragraph [0276]. Unfortunately, there are at least two problems with the Office's interpretation here.

[0051] First, there is nothing indicating that the event-related portions of Zintel cited by the Office (paragraphs [0273-0277]) are related to Zintel's message with property tags (from paragraphs [0436-0452]). The claim language states "customizable tag-based documents [that] holds events...." Since the Office equates the claimed "documents" to Zintel's messages (from paragraphs [0436-0452]), there must be and the Office must show, at the very least, how Zintel's messages are associated with the event-related portions of Zintel cited by the Office (paragraphs [0273-0277]). The Office has not done that. Applicant submits that Zintel's messages have nothing to do with the event-related portions of Zintel cited by the Office (paragraphs [0273-0277]).

[0052] Second, Zintel is completely silent about whether its events ("UPnP event notification") are or are not yet "consumed by an application service," which is part of

the recitation of this claim. Furthermore, the Office has not explained or shown how Zintel's events qualify as being "not yet consumed by an application service."

[0053] In addition, the Office indicates that Zintel's message with property tags (which it equated with the claimed "customizable tag-based documents") "holds the state [of] the one or more services." (Office Action, page 3). The Office supports that assertion by saying that Zintel's "UPnP events are mainly used for asynchronous notifications of state changes; paragraphs 0125 and 0294]." (Office Action, page 3). Those cited sections of Zintel are provided in paragraph [0014] of this response.

[0054] It appears the Office is equating the "customizable tag-based document that holds a state of the service," as recited in part by claim 21, with the, "UPnP events are mainly used for asynchronous notifications of state changes," and "Event-message generated when a change in a resource's state occurs." (Zintel, paragraphs [0125] and [0294]).

[0055] However, the Office has failed to show how Zintel's message with property tags "hold[s] a state of the service." Applicant submits that there is nothing indicating that these state-related portions of Zintel cited by the Office (paragraphs [0125] and [0294]) are related to Zintel's message with property tags (Zintel, paragraphs [0436-0452]). Since the Office equates the claimed "document" to Zintel's messages (Zintel, paragraphs [0436-0452]), there must be and the Office must show, at the very least, how Zintel's messages are associated with the state-related portions of Zintel cited by the Office (paragraphs [0125] and [0294]). The Office has not done that. Applicant submits that Zintel's messages have nothing to do with the state-related portions of Zintel cited by the Office.

[0056] In addition, claim 21 recites in part, "requesting the service to change a cursor shape." The Office states, "As to claim 21, see the rejection to claim 16 above." (Office Action, page 13). In its rejection of claim 16, the Office states, "Zintel and Slaughter does [sic] not specifically disclose requesting the service to change a cursor shape, the act of requesting invoking a cursor shape service that changes the shape of the cursor." (Office Action, page 12). Instead, the Office cites Saint, paragraphs [0144]-[0146], [0152] and [0155] as teaching this feature. (Office Action, page 12). Saint describes, "various types of input, from no input at all to pen-input, keyboard, mouse, and button input. Each type of input will generally be handled by a respective input service, wherein the existence and capabilities of each input service will be described in that service's UPnP description information." (Saint, paragraph [0144]).

[0057] It appears the Office is equating "requesting the service to change a cursor shape," as recited in part by claim 21, with the "respective input service" of Saint's paragraph [0144]. This is not the same as the "respective input service" described by Saint in paragraph [0144], because Saint's "respective input service" does not disclose, teach or suggest the claimed, "invoking a cursor shape service that changes the shape of the cursor, the cursor shape described by a customizable tag-based document, the customizable tag-based document including image data specifying pixels that comprise the cursor shape."

[0058] Instead, Saint describes, "three pointer primitives...(down, move, up)...used is [sic] almost all operating systems. If the device has a display, these pointer primitives may have a different resolution that may not match the x and y of the display...software residing on the extended PC may be used to scale the pointer position to the display."

(Saint, paragraphs [0152]-[0155]). Pointer primitives describe actions or movements of a pointer such as "down," "move," or "up." Saint's "pointer primitives" do not disclose, teach or suggest the claimed "describe… the cursor shape," or "include[es] image data specifying pixels that comprise the cursor shape." Instead, pointer primitives only seem to address the "down", "move" and "up" aspect of the pointer.

[0059] Furthermore, Saint fails to disclose, teach or suggest a "customizable tag-based document," as recited in part by amended claim 21. In general, Saint describes, "software residing on the extended PC may be used to scale the pointer position to the display." The PC software used to "scale the pointer" is not the same as the "customizable tag-based document" of amended claim 21, as "the cursor shape [is] described by a customizable tag-based document," and "include[es] image data specifying pixels that comprise the on-screen cursor." In contrast, Saint describes, "If the device has a display, these pointer primitives may have a different resolution that may not match the x and y of the display. In such instances, software residing on the extended PC may be used to scale the pointer position to the display." (Saint, paragraph [0155]). Scaling the pointer position to the display, as described by Saint, is not the same as "the cursor shape described by a customizable tag-based document," as claim 21 recites in part. Instead, scaling the pointer occurs when the pointer primitives (i.e. "down", "move" and "up") have a different resolution that does not match the x and y of the device's display.

[0060] Consequently, the combination of Zintel and Slaughter does not teach or suggest all of the elements and features of this claim. Accordingly, Applicant respectfully requests that the rejection of this claim be withdrawn.

Claims 13 and 15 Are Non-Obvious Over Zintel, Saint and Slaughter

[0061] Claims 13 and 15 stand rejected under 35 U.S.C. § 103(a) as allegedly being obvious over Zintel and Saint and further in view of Slaughter. Applicant respectfully traverses the rejection.

Dependent Claims 13 and 15

[0062] Claims 13 and 15 ultimately depend from independent claim 6. As discussed above, claim 6 is allowable over the cited documents. Therefore, claims 13 and 15 are also allowable over the cited documents of record for at least their dependency from an allowable base claim. These claims may also be allowable for the additional features that each recites.

Claims 17, 18, 20, 22 and 23 Are Non-Obvious Over Zintel, Slaughter and Saint

[0063] Claims 17, 18, 20, 22 and 23 stand rejected under 35 U.S.C. § 103(a) as allegedly being obvious over Zintel and Slaughter and further in view of Saint. Applicant respectfully traverses the rejection.

Dependent Claims 17, 18 and 20

[0064] Claims 17, 18 and 20 ultimately depend from independent claim 16. As discussed above, claim 16 is allowable over the cited documents. Therefore, claims 17,

18 and 20 are also allowable over the cited documents of record for at least their dependency from an allowable base claim. These claims may also be allowable for the additional features that each recites.

Dependent Claims 22 and 23

[0065] Claims 22 and 23 ultimately depend from independent claim 21. As discussed above, claim 21 is allowable over the cited documents. Therefore, claims 22 and 23 are also allowable over the cited documents of record for at least their dependency from an allowable base claim. These claims may also be allowable for the additional features that each recites.

Conclusion

[0066] Applicant submits that all pending claims are in condition for allowance. Applicant respectfully requests reconsideration and prompt issuance of the application. If any issues remain that prevent issuance of this application, the Examiner is urged to contact the undersigned representative for the Applicant before issuing a subsequent Action.

Respectfully Submitted,

Lee & Hayes, PLLC
Representative for Applicant

/kaseychristie40559/ Dated: 02/01/2010
Kasey C. Christie
(kasey@leehayes.com; 509-944-4732)
Registration No. 40,559